Large-scale Data Systems

Lecture 1: Introduction

Prof. Gilles Louppe
g.louppe@uliege.be

LIÈGE université

The zettabyte era

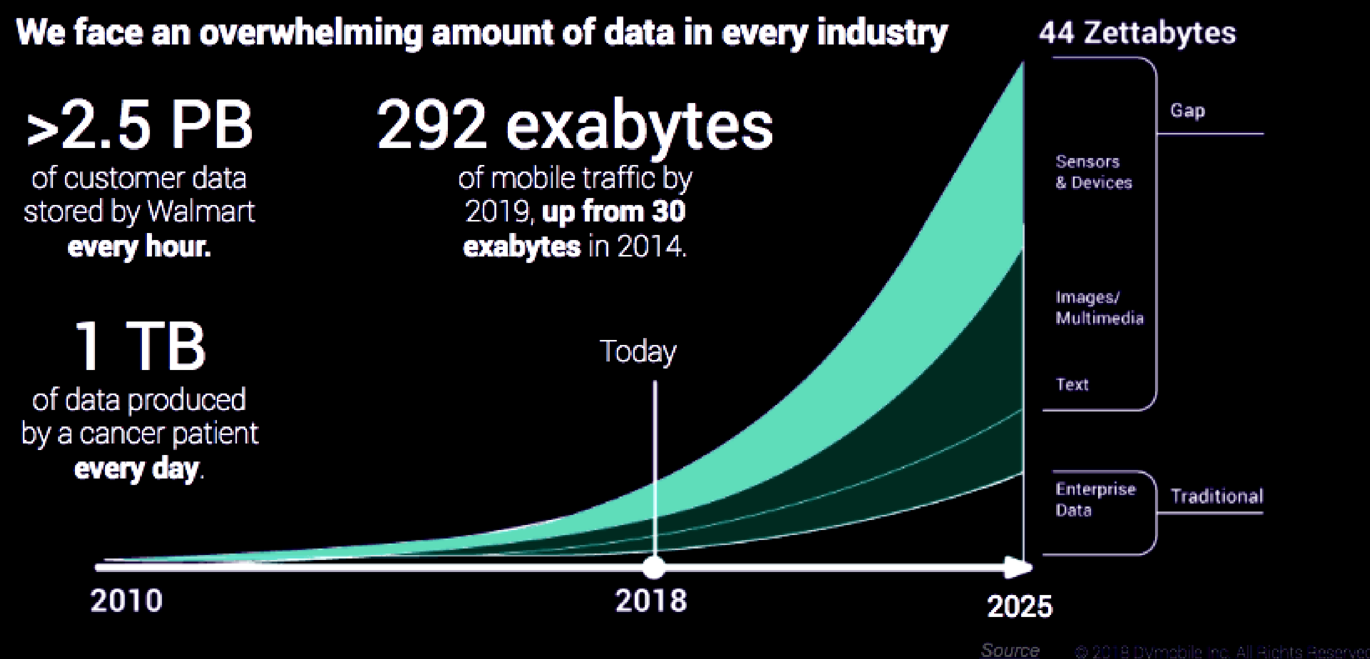
"Every two days now we create as much information as we did from the dawn of civilization up until 2003", according to Schmidt. That's something like five exabytes of data, he says.

Let me repeat that: ***we create as much information in two days now as we did from the dawn of man*** *through 2003.*

Eric Schmidt, 2010.


DATA
DATA EVERYWHERE

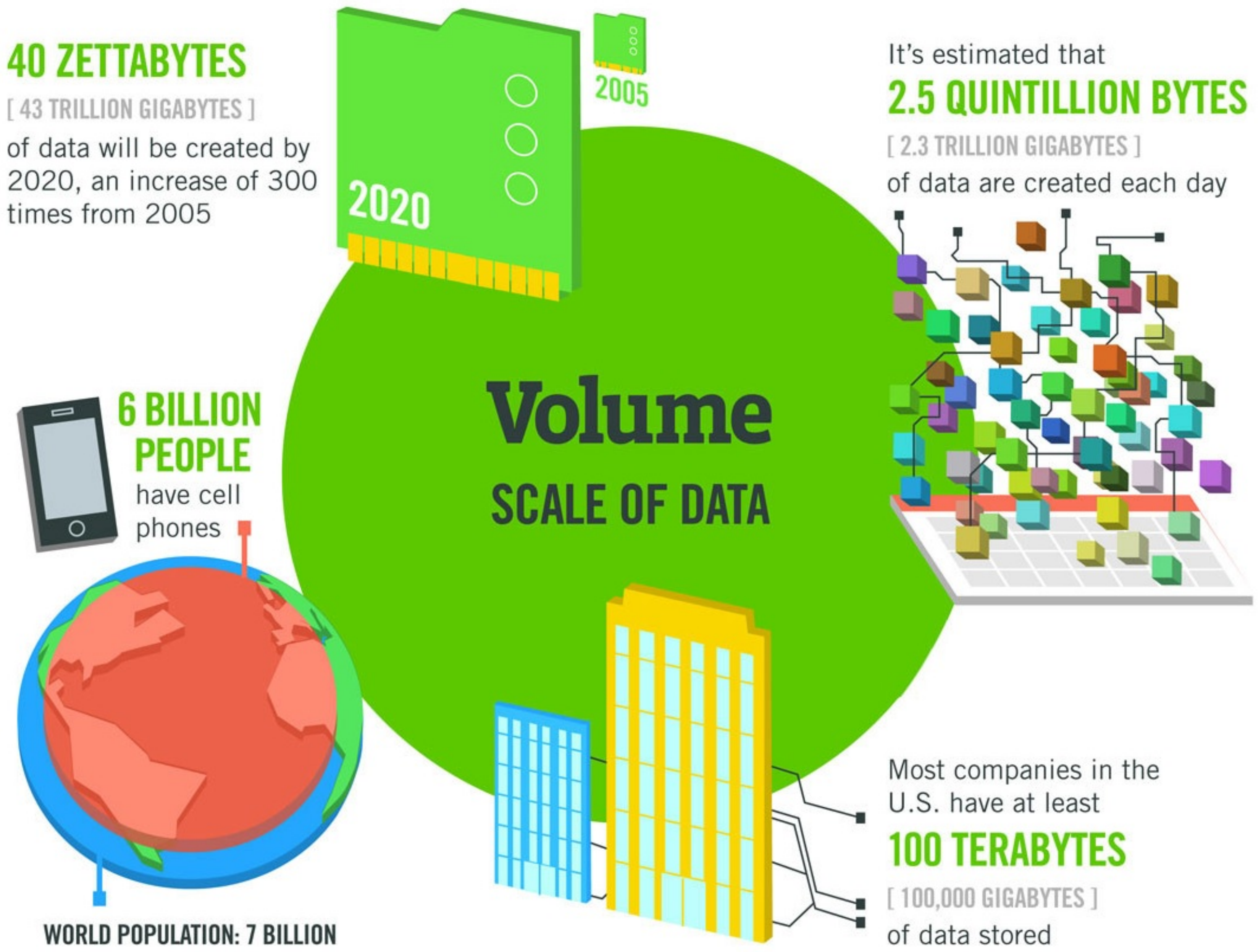
1 Zettabyte (ZB) = 1 Trillion Gigabytes (GB)

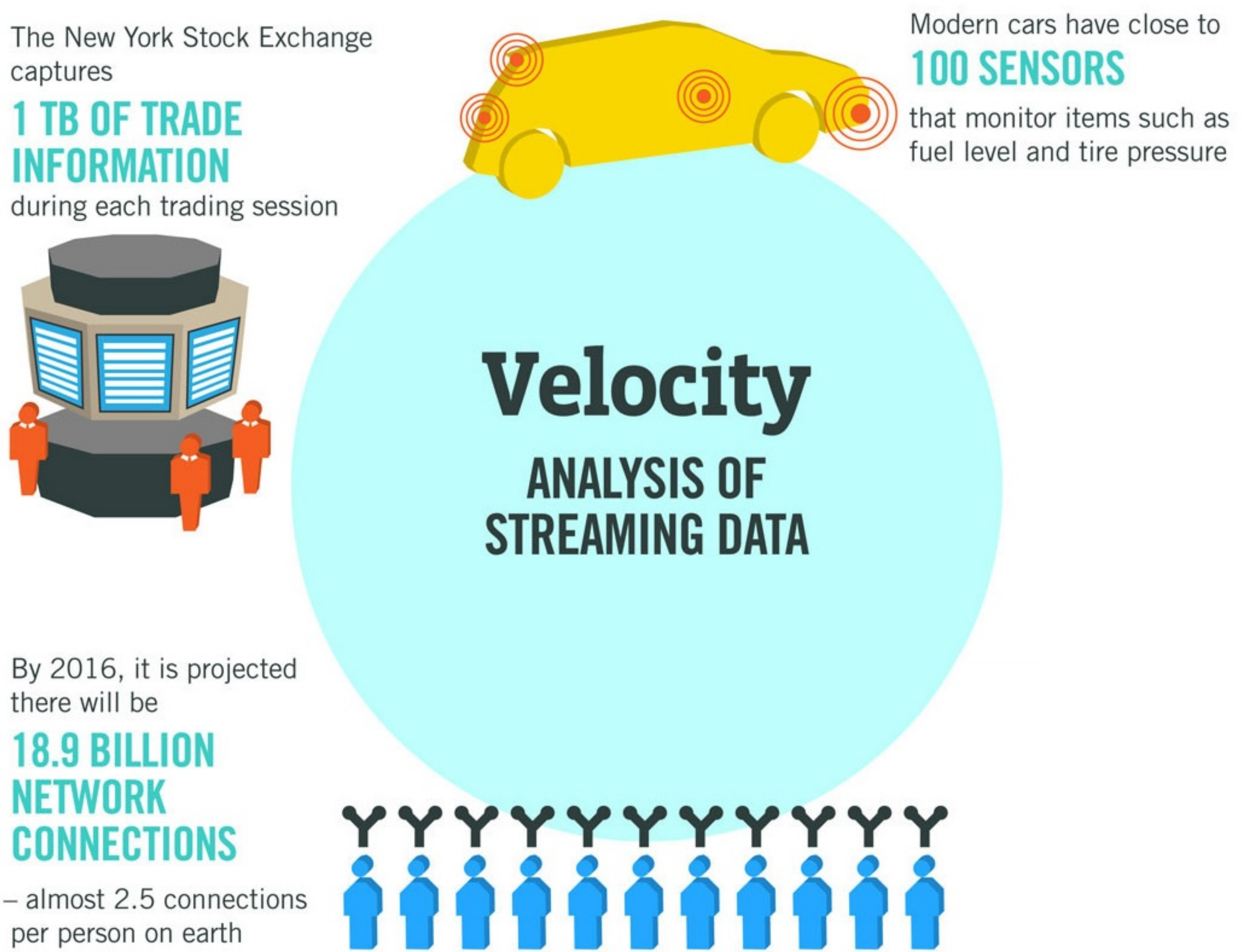


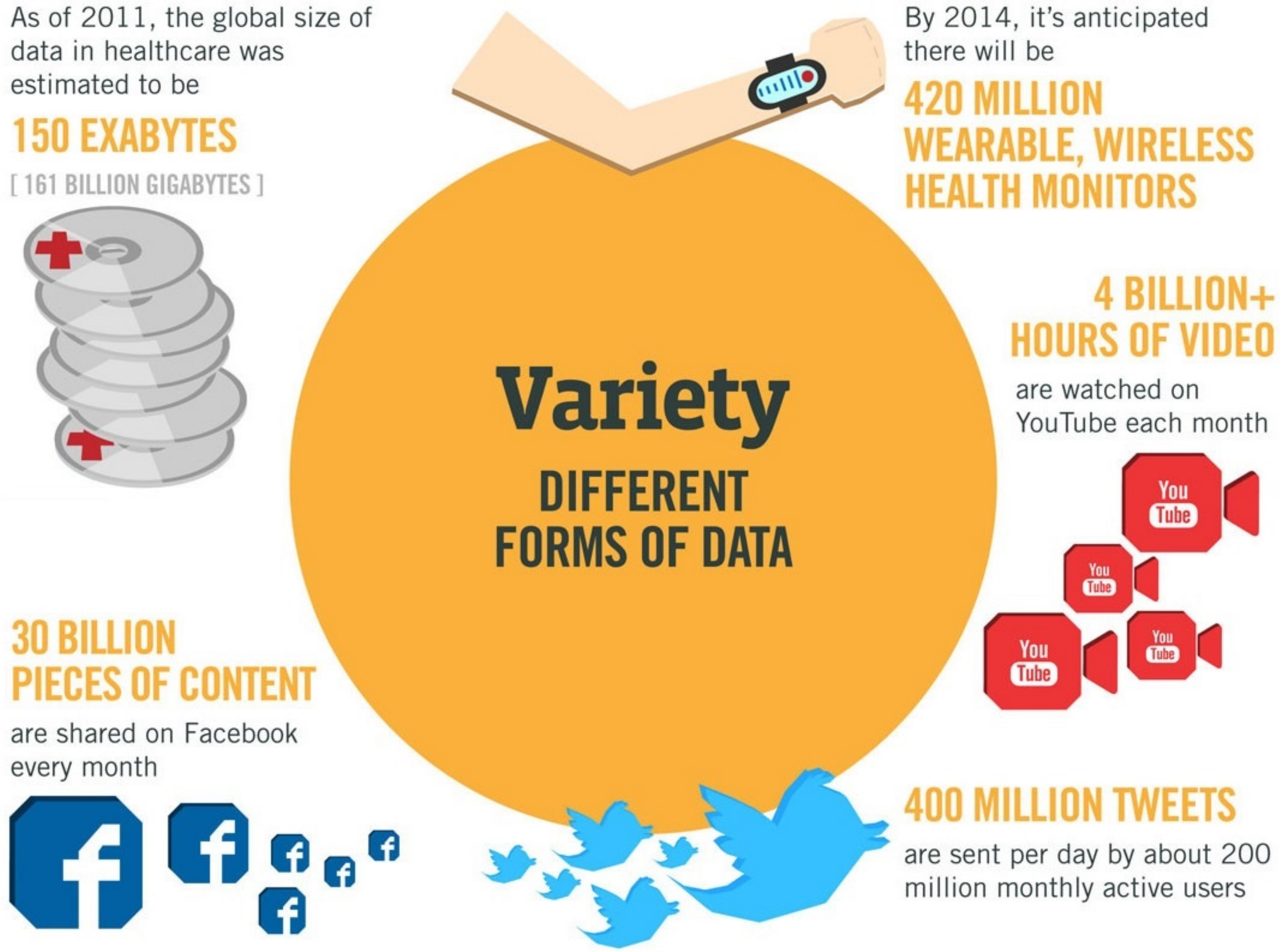
Exercise

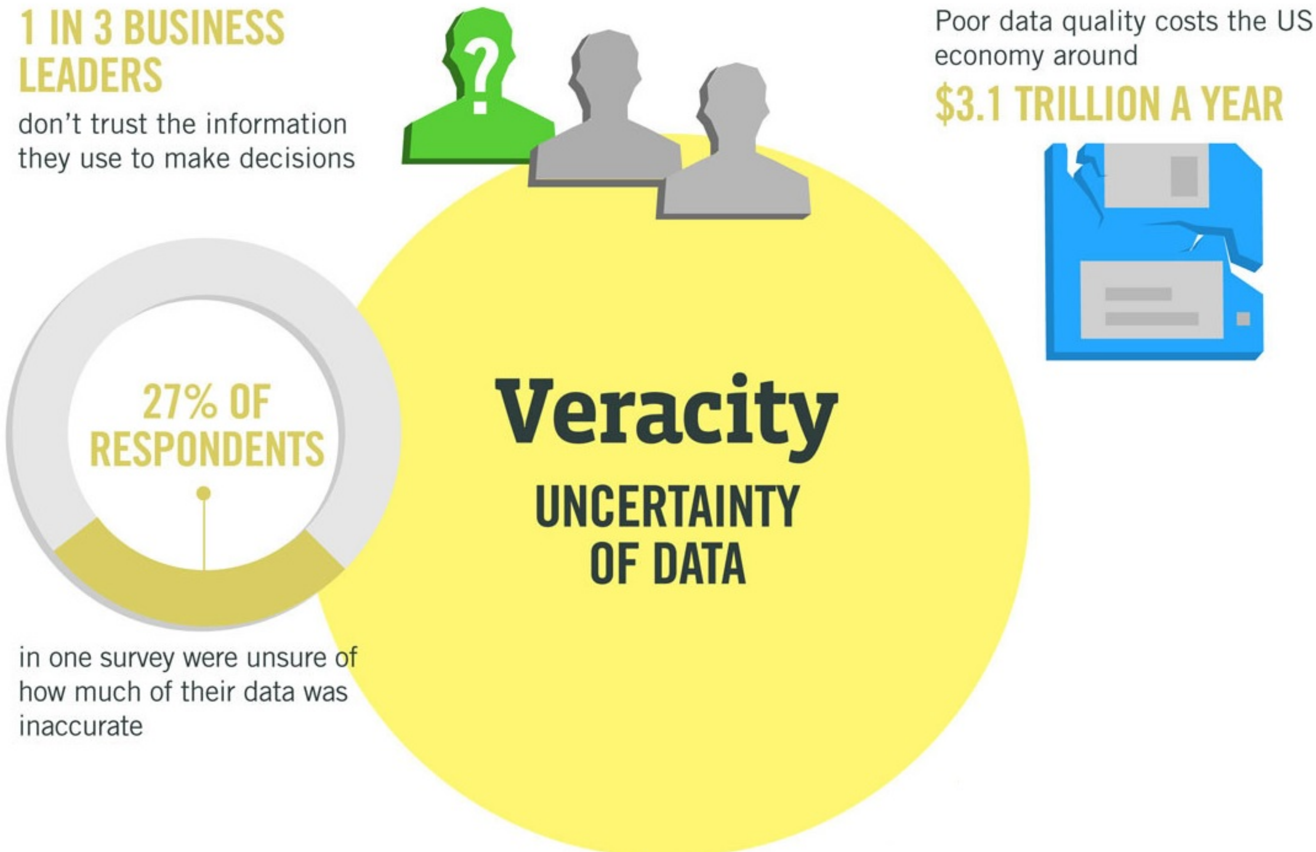
- How many iPads would you need to store 1 ZB?
- What distance and volume does that represent?

The four V of big data




The New York Stock Exchange captures
1 TB OF TRADE INFORMATION
during each trading session
Modern cars have close to
100 SENSORS
that monitor items such as fuel level and tire pressure
Velocity
ANALYSIS OF STREAMING DATA
By 2016, it is projected there will be
18.9 BILLION NETWORK CONNECTIONS
– almost 2.5 connections per person on earth


As of 2011, the global size of data in healthcare was estimated to be
150 EXABYTES
[161 BILLION GIGABYTES]
By 2014, it's anticipated there will be
420 MILLION WEARABLE, WIRELESS HEALTH MONITORS
Variety
DIFFERENT FORMS OF DATA
4 BILLION+ HOURS OF VIDEO
are watched on YouTube each month
30 BILLION PIECES OF CONTENT
are shared on Facebook every month
400 MILLION TWEETS
are sent per day by about 200 million monthly active users


1 IN 3 BUSINESS LEADERS
don't trust the information they use to make decisions
Poor data quality costs the US economy around
$3.1 TRILLION A YEAR
27% OF RESPONDENTS
in one survey were unsure of how much of their data was inaccurate
Veracity
UNCERTAINTY OF DATA

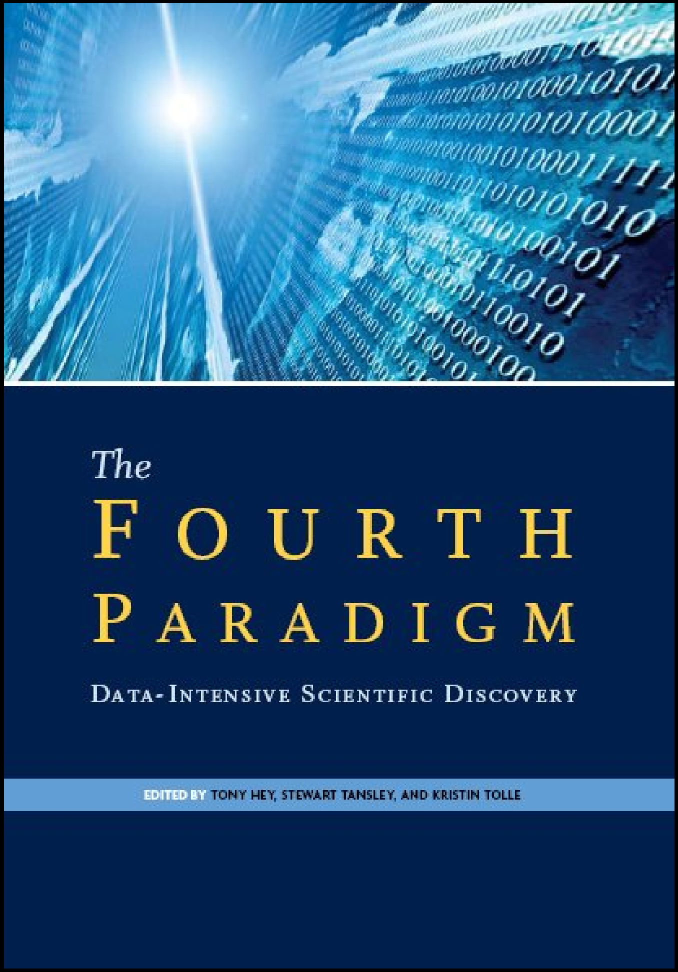
Example: your own Facebook data

How much data do you generate through your interactions with Facebook?

Let's check!

What's new?

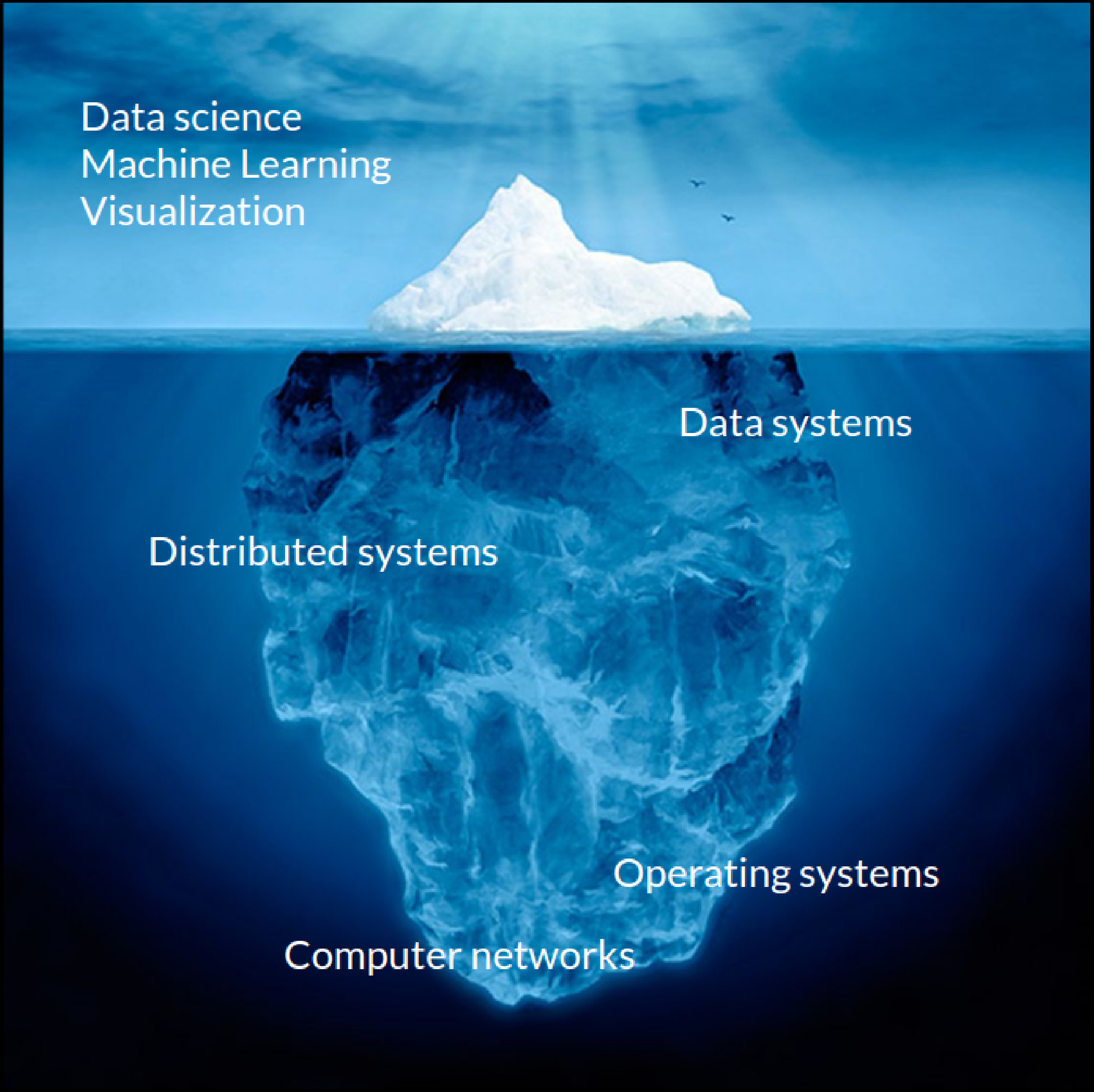
- our ability to store machine generated data, at unprecedented scale and rate.
- the broad understanding that we cannot just manually get value out of data.



"Increasingly, scientific breakthroughs will be powered by advanced computing capabilities that help researchers manipulate and explore massive datasets.

The speed at which any given scientific discipline advances will depend on how well its researchers collaborate *with one another, and with technologists, in areas of eScience such as* databases, workflow management, *visualization, and* cloud computing technologies.*"*

Data systems


Data science
Machine Learning
Visualization
Data systems
Distributed systems
Operating systems
Computer networks

Operating systems

Can you name examples of operating systems?

Operating systems

Can you name examples of operating systems?

- Android
- Chrome OS
- FreeBSD
- iOS
- macOS
- OS/2
- RISC OS
- Solaris
- Windows
- ...

Definition

The low-level software which handles the interface to peripheral hardware, schedules tasks, allocates storage, and presents a default interface to the user when no application program is running.

Distributed systems

Can you name examples of distributed systems?

Distributed systems

Can you name examples of distributed systems?

- A client/server system
- The web
- Wireless networks
- Telephone networks
- DNS
- Massively multiplayer online games
- Distributed databases
- BitTorrent (peer-to-peer overlays)
- A cloud, e.g. Amazon EC2/S3, Microsoft Azure
- A data center, e.g. a Google data center, AWS
- The bitcoin network

Definition

A distributed system is a collection of entities with a common goal, each of which is autonomous, programmable, asynchronous and failure-prone, and which communicate through an unreliable communication medium.

- Entity: a process on a device.
- Communication medium: Wired or wireless network.

A distributed system appears to its users as a single coherent system.

Internet: What are the entities? What is the communication medium?

Data center: What are the entities? What is the communication medium?

Why study distributed systems?

- Distributed systems are everywhere:
  - Internet
  - WWW
  - Mobile devices
  - Internet of Things
- Technical importance:
  - Improve scalability
    - Adding computational resources to a system is an easy way to scale its performance to many users.
  - Improve reliability
    - We want high availability and durability of the system.

- Distributed systems are difficult to build.
  - Scale: hundreds or thousands of machines.
    - Google: 4k-machine MapReduce cluster
    - Facebook: 60k machines providing the service
  - Fault tolerance: machines and networks do fail!
    - 50 machine failures out of 20k machine cluster per day (reported by Yahoo!)
    - 1 disk failure out of 16k disks every 6 hours (reported by Google)
  - Concurrency:
    - Nodes execute in parallel
    - Messages travel asynchronously
  - Consistency:
    - Distributed systems need to ensure user guarantees about the data they store.
    - E.g., all read operations return the same value, no matter where it is stored.
- But only a few core problems reoccur.

Teaser: Two Generals' Problem

Two generals need to coordinate an attack.

- They must agree on time to attack.
- They will win only if they attack simultaneously.
- They communicate through messengers.
- Messengers may be killed on their way.

Exercise
What should they do?

The Two Generals Problem

Later bekijk...

Delen

Let's try to solve the problem for generals g_1 and g_2.

- g_1 sends time of attack to g_2.
- Problem: how to ensure g_2 received the message?
- Solution: let g_2 acknowledge receipt of message.
- Problem: how to ensure g_1 received the acknowledgment?
- Solution: let g_1 acknowledge receipt of acknowledgment.
- ...

This problem is provably impossible to solve!

(Unless we make additional assumptions)

- Applicability to distributed systems:
    - Two nodes need to agree on a value.
    - They communicate by messages using an unreliable channel.
- Agreement is one of the core problems of distributed systems.

Data systems

Can you name examples of data systems?

Data systems

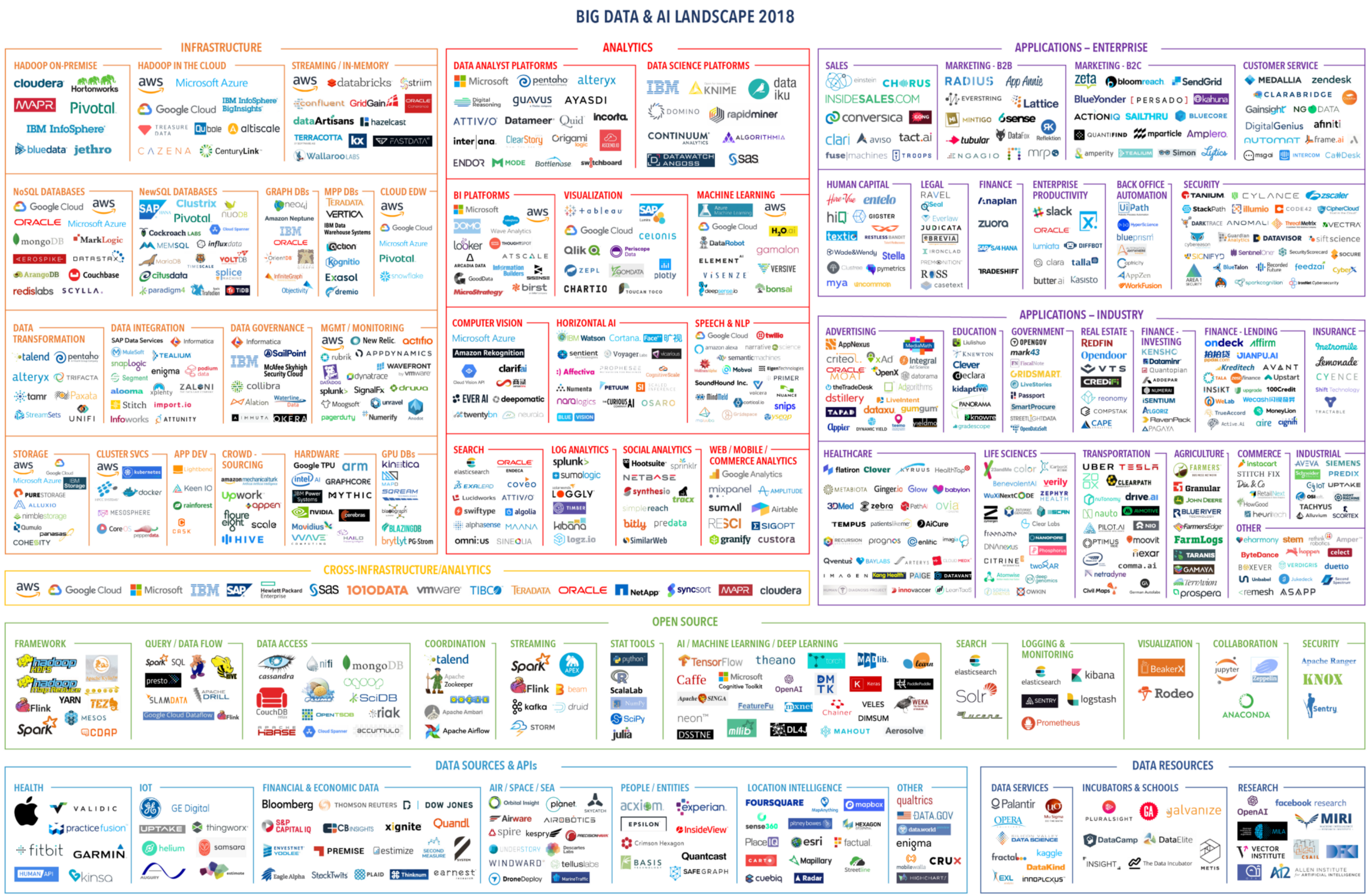
Can you name examples of data systems?

- A database
- A file system
- A ledger
- Search engines
- Data flow frameworks
- Social networks
- ...

Definition

In this course, data systems will broadly refer to any kind of computer systems, distributed or not, that can be used to store, retrieve, organize or process data.

Our main focus will be on distributed data systems.

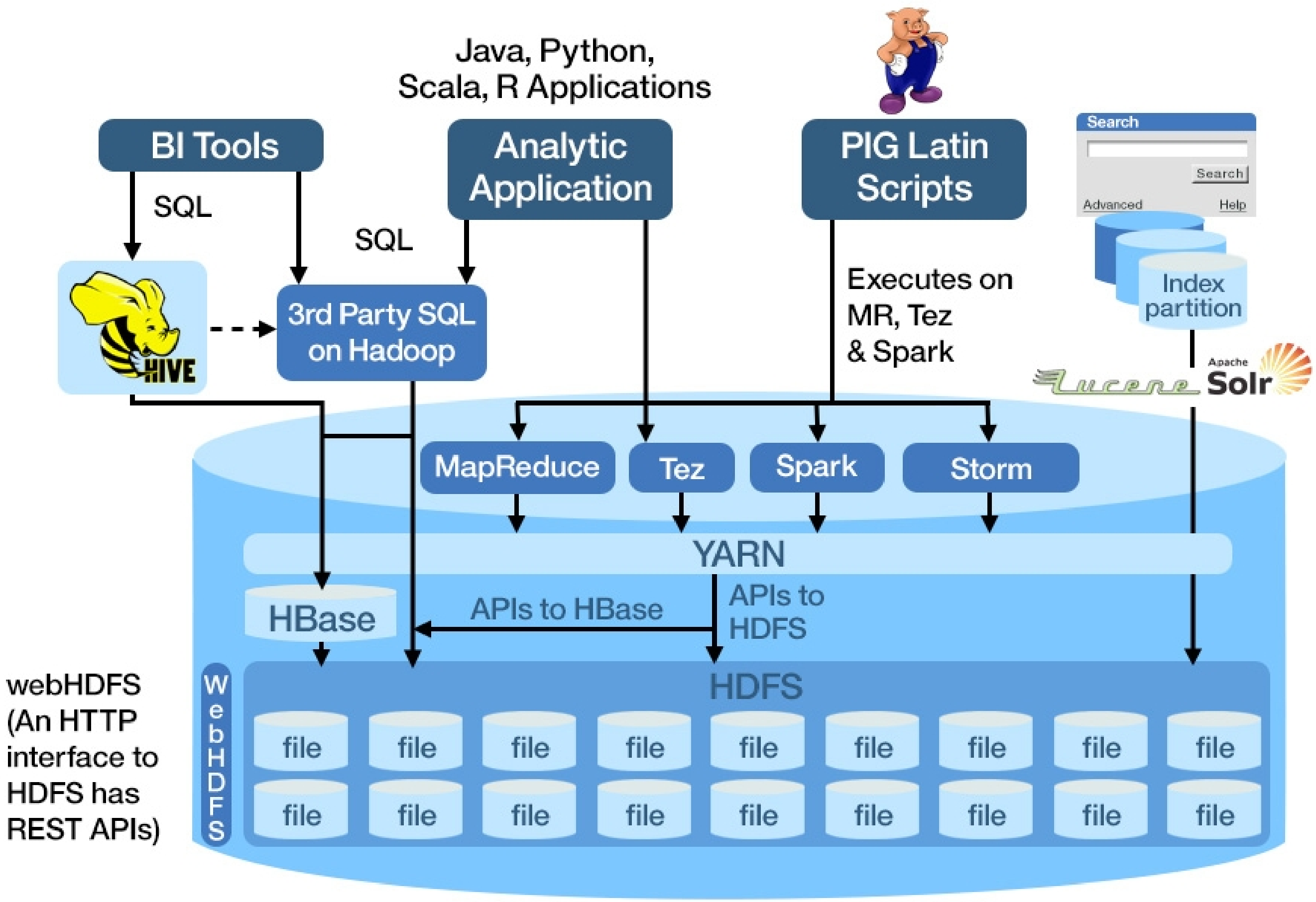


Final 2018 version, updated 07/15/2018 © Matt Turck (@mattturck), Demi Obayomi (@demi_obayomi), & FirstMark (@firstmarkcap) mattturck.com/bigdata2018

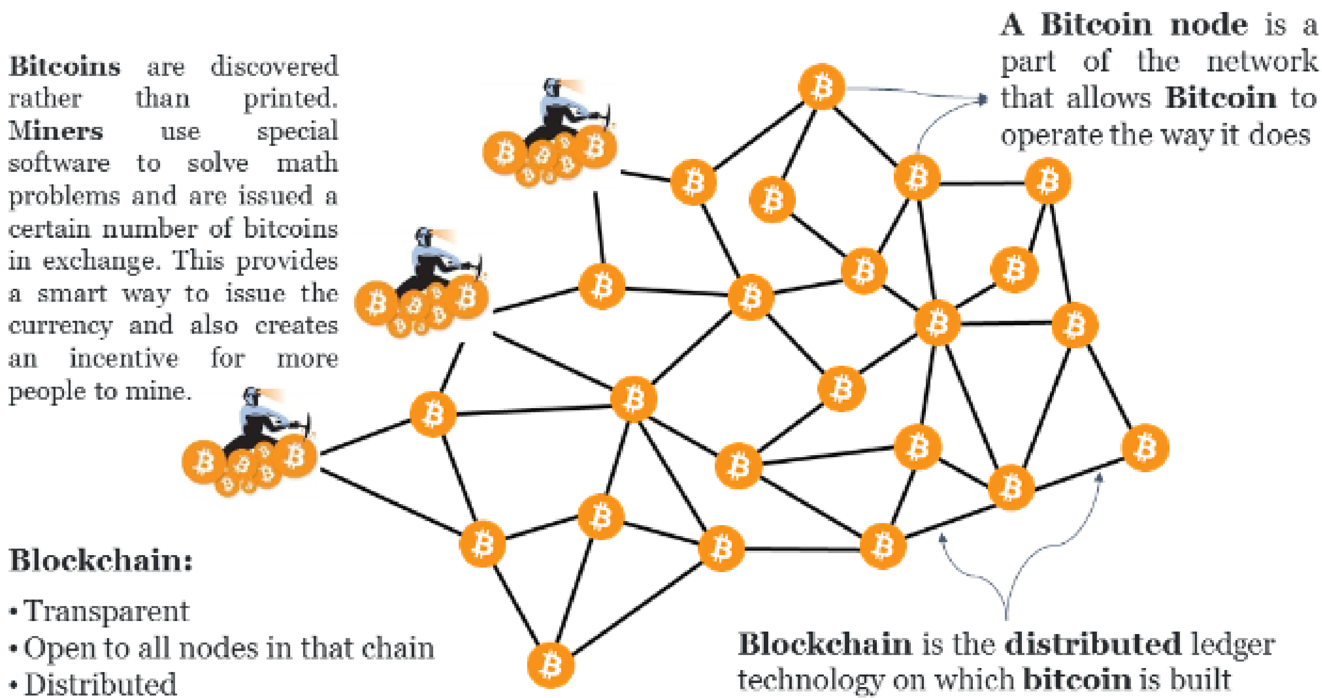
FIRSTMARK
EARLY STAGE VENTURE CAPITAL

Cloud services

The Hadoop ecosystem (less and less)



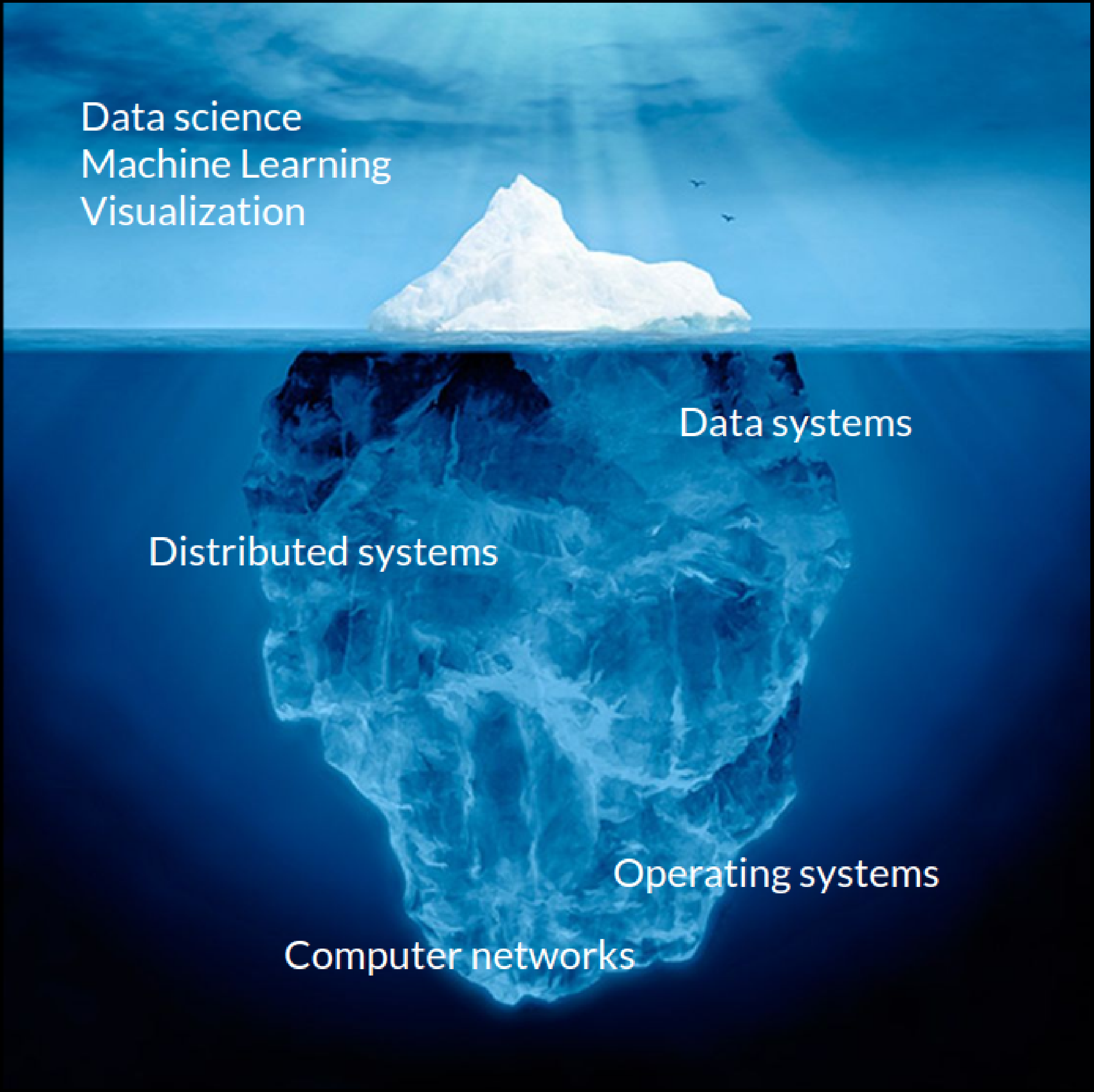
A distributed ledger

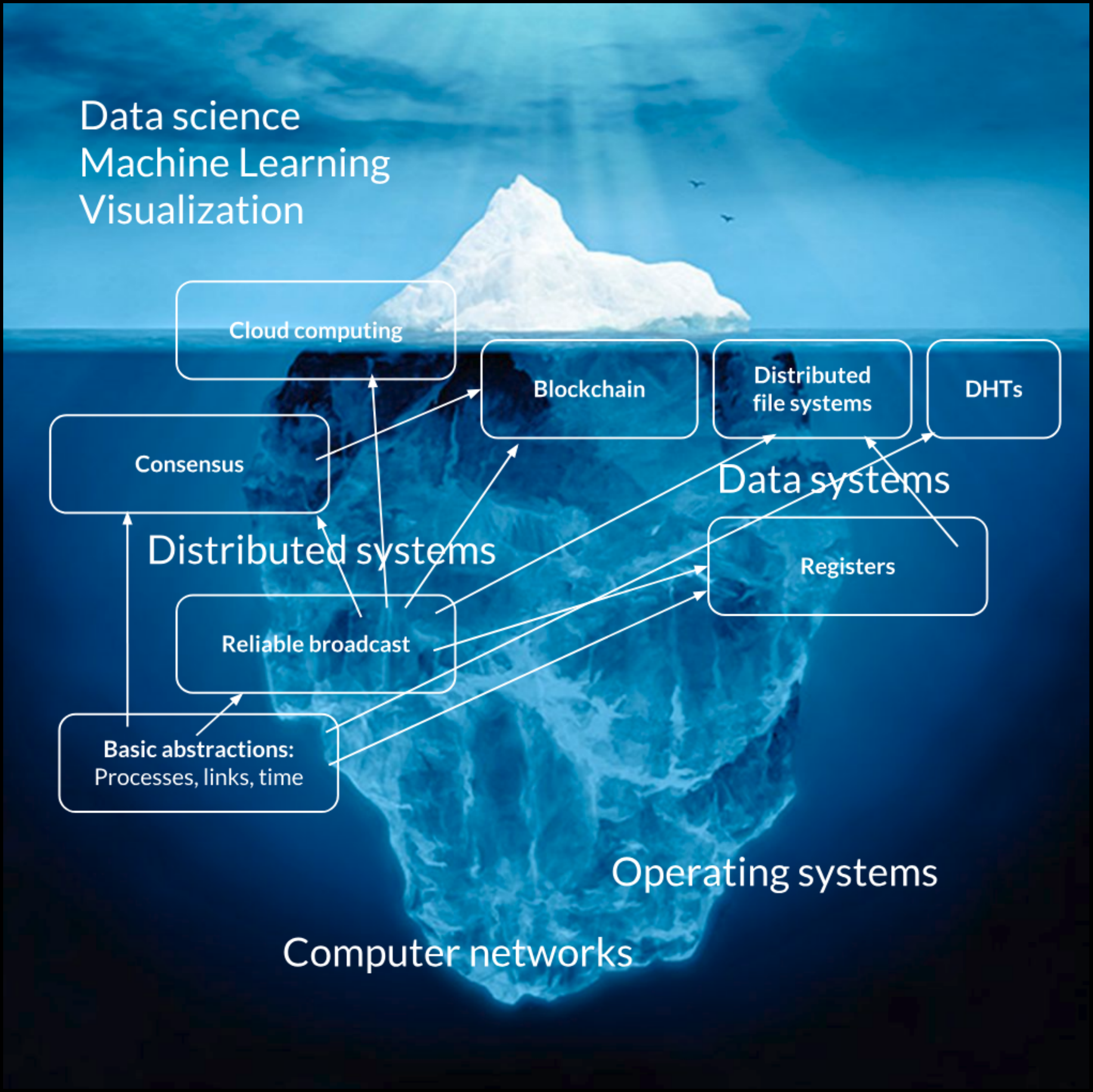


Outline

Outline

- Lecture 1: Introduction
- Lecture 2: Basic distributed abstractions
- Lecture 3: Reliable broadcast
- Lecture 4: Shared memory
- Lecture 5: Consensus
- Lecture 6: Distributed file systems
- Lecture 7: Distributed hash tables
- Lecture 8: Blockchain
- Lecture 9: Distributed computing


Data science
Machine Learning
Visualization
Data systems
Distributed systems
Operating systems
Computer networks


Data science
Machine Learning
Visualization
Cloud computing
Blockchain
Distributed file systems
DHTs
Consensus
Data systems
Distributed systems
Registers
Reliable broadcast
Basic abstractions:
Processes, links, time
Operating systems
Computer networks

Fundamentals of distributed systems

Understand the foundational principles required for the design, implementation and maintenance of distributed systems.

- Communications
- Failures
- Consistency
- Concurrency
- Consensus

Communications

- How do you talk to another machine?
  - Reliable networking.
- How do you talk to multiple machines at once, with ordering guarantees?
  - Multicast, Gossiping.

Failures and consistency

- How do you know if a machine has failed?
  - Failure detection.
- How you do program your system to operate continually even under failure?
  - Gossiping, replication.
- What if some machines do not cooperate?
  - Byzantine fault tolerance.

Concurrency

- How do you control access to shared resources?
  - Distributed mutual exclusion, distributed transactions, etc.

Consensus

- How do multiple machines reach an agreement?
  - Time and synchronization, global states, leader election, Paxos, proof of work, blockchain.
- Bad news: it is impossible!
  - The impossibility of consensus for asynchronous systems.

Case studies

From these building blocks, understand how to build and architecture data systems for large volumes or data or for data science purposes.

Distributed storage

- How do you locate where things are and access them?
    - Distributed file systems
    - Key-value stores
- How do you record and share sensitive data?
    - Proof of work, blockchain

Distributed computing for data science

- What are the distributed computing systems for data science?
  - Map Reduce (Hadoop)
  - Computational graph systems (Spark, Dask)
- Is distributed computing always necessary?

We give you Lego bricks. Your job is to assemble them into well-engineered solutions.

—
Image credits: Kelly Sikkema.

The end.